

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平7-212397

(43)公開日 平成7年(1995)8月11日

(51)Int.Cl.⁶
H 04 L 12/56識別記号
9077-5K

F I

技術表示箇所

H 04 L 11/ 20

1 0 2 D

審査請求 有 請求項の数10 O L (全 21 頁)

(21)出願番号 特願平6-288309
 (22)出願日 平成6年(1994)11月22日
 (31)優先権主張番号 93480231.5
 (32)優先日 1993年12月22日
 (33)優先権主張国 フランス (F R)

(71)出願人 390009531
 インターナショナル・ビジネス・マシーンズ・コーポレーション
 INTERNATIONAL BUSINESSES MACHINES CORPORATION
 アメリカ合衆国10504、ニューヨーク州
 アーモンク (番地なし)
 (72)発明者 ベルタン・オリヴィエ
 フランス06200 ニース ブールヴァール・ド・モンレアル 53
 (74)代理人 弁理士 合田 澄 (外2名)

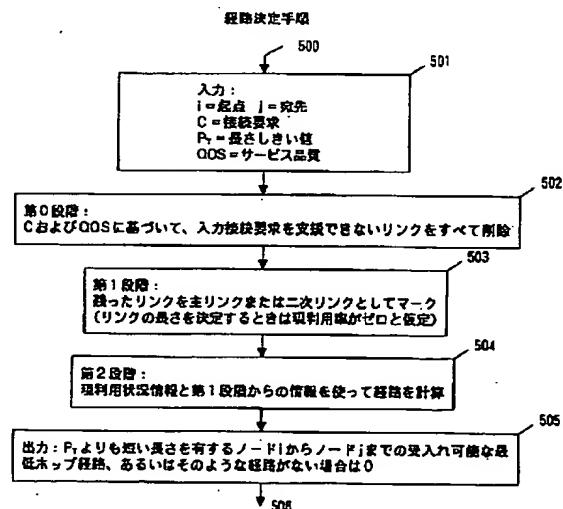
最終頁に続く

(54)【発明の名称】 最適経路を決定するための方法及びネットワーク・ノード

(57)【要約】

【目的】 経路指定アルゴリズムによって調査される可能性のあるリンクの総数を減らすことによって、パケット交換ネットワークにおいて起点ノードと宛先ノードの間の指定経路をより効率的に選択する。

【構成】 この経路選択技法は、主リンクを識別するための第1段階と、ネットワークにおける最適の2地点間経路を選択するための第2段階の2つの段階を含む。主経路は、指定されたしきい値よりも少ない伝送遅延を有する最低ホップ・カウント経路である。第2段階において、ネットワーク・リンクのスクリーニングを主リンクのみに制限し、二次リンクを除外する。調査するリンクの数をより少なくすると、経路選択に必要とされる計算時間がかなり減少するという利点がある。一般に、主リンクの数は、潜在的に調査される可能性のあるリンクの総数に比べて少ないため、試みが成功しなかった場合に必要な追加の処理時間が制限できる。



【特許請求の範囲】

【請求項1】 伝送リンク(209)と相互接続された複数のノード(201～208)を含むパケット交換ネットワーク(200)におけるネットワーク・ノード(300)であって、データ・パケットを受信し送信する手段(301、302、304)と、ネットワーク構成を記憶し更新する手段(306)と、前記ネットワークにおける起点ノードと宛先ノードの間において、最低ホップ・カウントと最低伝送遅延時間Σdklを有する主経路を識別し、主リンクと二次リンクを識別する手段と、主経路と主リンクを識別する前記手段に応答して、前記起点ノードと宛先ノードの間の最適経路を決定する手段とを含み、前記最適経路を決定する前記手段が、主リンクだけを使って最適経路を決定する手段を含むことを特徴とするネットワーク・ノード。

【請求項2】 前記最適経路を決定する前記手段が、前記主リンクだけを使って宛先ノードに到達しないときは、主リンクおよび二次リンクを使って最適経路を決定する手段を含む、請求項1に記載のネットワーク・ノード。

【請求項3】 主リンクおよび二次リンクを使って最適経路を決定する前記手段が、前記主リンク上で最適経路を決定する前記手段によって主リンク上で以前に計算済みのリンク使用状況データを利用することを特徴とする、請求項2に記載のネットワーク・ノード。

【請求項4】 前記最適経路における伝送遅延を、事前選択したしきい値PTに制限する手段を含むことを特徴とする、請求項1ないし3のいずれか1に記載のネットワーク・ノード。

【請求項5】 前記起点ノードと宛先ノードの間の最適経路を決定する前記手段が、修正ベルマン・フォード・アルゴリズムを実行する手段を含む、請求項1ないし4のいずれか1に記載のネットワーク・ノード。

【請求項6】 伝送リンク(209)と相互接続された複数のノード(201～208)を含むパケット交換通信ネットワーク(200)におけるアクセス・ノード(300)において、起点ノードと宛先ノードの間の最適経路を決定するための方法であって、ネットワーク構成を記憶し更新するステップと、

第1段階(503)で、前記ネットワークにおいて前記起点ノードと前記宛先ノードの間において最少ホップ・カウントと最低伝送遅延時間Σdklを有する主経路を識別し、主リンクと二次リンクを識別するステップと、第2段階(504)で、主経路と主リンクを識別する前記ステップに応答して、前記起点ノードと宛先ノードの間で最適経路を決定するステップを含み、最適経路を決定する前記第2段階(504)が、主リンクだけを使って最適経路を決定するステップ(701、

704)を含むことを特徴とする方法。

【請求項7】 最適経路を決定する前記第2段階(504)が、前記主リンクだけを使って宛先ノードに到達しないとき(702)は、主リンクおよび二次リンクを使って最適経路を決定するステップ(703、704)を含む、請求項6に記載の方法。

【請求項8】 主リンクおよび二次リンクを使って最適経路を決定する前記ステップが、前記主リンク上で最適経路を決定する前記ステップによって主リンク上で以前に計算済みのリンク使用状況データを利用することを特徴とする、請求項7に記載の方法。

【請求項9】 前記最適経路における伝送遅延を、事前選択したしきい値PTに制限するステップ(505)を実行するステップを含む、請求項6ないし8のいずれか1に記載の方法。

【請求項10】 前記起点ノードと宛先ノードの間の最適経路を決定する前記ステップが、修正ベルマン・フォード・アルゴリズムを含むことを特徴とする、請求項6ないし9のいずれか1に記載の方法。

20 【発明の詳細な説明】

【0001】

【産業上の利用分野】 本発明は、大規模パケット交換ネットワークにおける高速伝送システムに関し、より詳細には、修正ベルマン・フォード・アルゴリズムによって調査される可能性のあるリンクの総数を減らすことにより、ソース・ノードと宛先ノードの間の経路選択の速度を速めるための装置および方法に関する。

【0002】

【従来の技術】 データを「パケット」と呼ばれるデータの集合体として伝送するパケット交換ネットワーク手段によって、複数のデータ処理要素を相互接続することができます有用になってきた。そのようなネットワークは、データ処理装置を支援するエンド・ノードに接続された複数の相互接続された交換ノードを含んでいる。そのようなパケット・ネットワークは、地理的に広く分布した非常に大きなものになり得る。そのような状況では、互いに通信しようとする2つのエンド・ノード間の効率的な経路選択が最も重要になる。H. ヌスバウマー(Nussbaumer)のTeleinformatique2 (pp. 92～117) "Press 30 es Polytechniques Romandes 1987" にさまざまな方法が要約されている。

【0003】 経路選択：ネットワーク・レベルで解決しなければならない一般的な問題は、ソース・ノードと宛先ノードの間の経路を見つけることである。ネットワークがデータグラムを使用するときは、経路選択は各パケットごとに個別に行わなければならない。仮想回線では、経路決定は回路確立時にだけ1度行われる。どちらの場合にも、経路指定アルゴリズムの選択は、しばしば相反する多くの必要条件を満たさなければならないの 40 で、容易ではない。このアルゴリズムは、ノードの実現

が複雑にならないように実施できるよう簡単でなければならず、ネットワーク上にどんな混乱があっても、正確なパケットの進行を保証しなければならない。トラフィックの変動やネットワーク・トポロジのいかんにかかわらず、このアルゴリズムは満足できる結果を提供できなければならない。また、さまざまなエンド・ユーザが公平になるようにし、ネットワークにアクセスする権利を平等に分配しなければならない。最後に、経路指定アルゴリズムは、できれば利用タイプによって変わり得る基準に従って、ネットワークを最適に利用できるようにしなければならない。多くの場合、ネットワークは、パケット伝送時間が最短になり、かつ最大数のパケットが転送できるように実施される。その主な目的は、それぞれ伝送時間の削減とスループットの向上である。他の場合には、通信コストを削減し、あるいは破局的な回線、ノード故障、トラフィック・ピークの場合にも正確に動作（過度に性能が低下することなく）できる信頼性の高いネットワークを開発することがその目的である。さまざまの制約があるため、多数の異なるタイプの経路指定がある。

【0004】制約条件が多様であるため、あふれ(floating)経路指定、ランダム経路指定または確率的(stochastic)経路指定、決定的経路指定など、異なる多数のタイプの経路指定がある。この最後の技法は、最も幅広く使われている技法の1つであり、固定式経路指定と適応的経路指定の対立する2つの動作モードがある。

【0005】(1) 固定式経路指定

固定式経路指定は、ネットワーク・トポロジや通信回線上で予想される平均トラフィックなど一般のネットワーク特性にしたがって、異なるパケットがとるべき経路を定義する。経路指定規則は1回確立され、ユーザが優先する性能基準の最適化を狙いとする。固定式経路選択は、ネットワークの長期的かつ総合的な最適化を狙いとするが、ランダム経路選択と比較すると、ネットワークを介するパケット伝送速度をかなり高めることができる。

【0006】(2) 適応的経路指定：

固定式経路指定とは反対に、適応的経路指定の目的はいかなるときも最適化基準を満たすことである。たとえば回線上のトラフィックの瞬間的状態に応じて、上述のテーブルが絶えず更新される。

【0007】・集中経路指定：

ネットワーク特性が絶えず変化するときは、トラフィックの変動およびトポロジに応じて経路指定テーブルを周期的に更新する責任をあるノードに割り当てることによって、経路指定を適応化することができる。この方法は、原理的には単純であり、集中経路指定と呼ばれる。その主な欠点は、大きな補助トラフィックを発生すること、およびネットワークの良好な機能をただ1つのノードに従属させてしまうことである。さらに、テーブルが

すべてのノードによって同時に受け取られることはないので、集中経路指定はテーブルがリフレッシュされるときにいくつかの問題を引き起こす可能性がある。1つの解決策は、テーブルの確立を各ノード・レベルに分散させることである。

【0008】・ローカル経路指定：

ローカル経路指定方法も分散式経路指定方法（後述）も共に、局的に収集された情報を従って、各ノードによってそれ自体の経路指定テーブルを確立することに基づくものである。ローカル経路指定技法では、各ノードが情報を隣接ノードと交換することなくそのテーブルを作成する。最も簡単な形では、この方法は、受け取ったばかりのパケットをより短い出力待ち行列内に置き、それをできるだけ迅速に伝送することからなる。ローカル・アルゴリズムは原理的に、ネットワーク内でパケットを非常に素早く循環させる傾向がある。ただし、経路はさまざまな方法で選択されるので、平均経路長は最小値とは大きく異なる。

【0009】・分散式経路指定：

分散式経路指定は、経路指定テーブルと遅延テーブルを更新するために、隣接ノードがトラフィックおよびネットワーク状態に関するメッセージを交換する局的経路指定方法である。

【0010】高性能ネットワーク：

データ伝送は、アプリケーションに特に焦点を合わせ、また顧客のトラフィック・プロファイルに基本シフトを組み込むことにより進展してきている。ワークステーション、ローカル・エリア・ネットワーク（LAN）相互接続、ワークステーションとスーパー・コンピュータの間の分散処理、新しいアプリケーション、および階層ネットワークと対等ネットワーク、広域ネットワーク（WAN）とローカル・エリア・ネットワーク、音声とデータなどしばしば相反する異なる構造の統合の成長により、データ・プロファイルは、帯域幅が大きくなり、バースト的、非確率論的になり、またより高い接続性を必要とする。上記のことから、チャネル接続されたホスト、事務およびエンジニアリング・ワークステーション、端末装置並びに小さいものでは中間ファイル・サーバの間でLAN通信、音声、ビデオおよびトラフィックを搬送できる高速ネットワークの分散計算アプリケーションを支援することが強く必要とされていることは明らかである。高速マルチプロトコル・ネットワークのこの展望が、データ、音声、およびビデオ情報がデジタル・コード化され、小さなパケットに分断され、共通の1組のノードとリンクを介して伝送される、高速パケット交換ネットワーク・アーキテクチャの出現の原動力となっている。

【0011】(a) スループット

これらの新しいアーキテクチャの重要な要件は、実時間配布の制約を満たし、音声およびビデオの移送に必要な

ノードの高スループットを達成するために、端末間の遅延時間を減少させることである。リンク速度の増大に比例して、通信ノードの処理速度が増大しないと、高速ネットワークの基本課題は、各ノード内のパケット処理時間を最小にすることである。例えば(図1)、1MIPS(毎秒100万回の命令)の有効処理速度を有するノードを取り上げてみると、処理されるパケットごとにネットワークが833000個の命令を実行しなければならない場合でも、9.6kbpsの回線を1000バイトのパケットで満たすことができる。64kbpsの回線では、ノードはパケットごとに125000回の命令を収容できる。しかしながら、OC24リンクを満たすためには、この1MIPSのノードは、パケットごとに7個の命令しか実行できない。後者の場合は、10~30MIPSの有効処理速度でも、パケットごとに70~200回の命令しか収容できない。処理時間を最小にし、高速/低エラー率の技法を完全に利用するために、新しい高帯域ネットワーク・アキテクチャによって提供される転送および制御機能の大部分は、端末間方式で実行される。フロー制御および特に経路選択は、中間ノードの知識と機能の両方を減少させるネットワークのアクセスポイントによって管理される。

【0012】(b) 経路指定モード

ネットワーク内の経路指定は、次の2つの側面をもつ。

1. 所与の接続のための経路を決定する。

2. 交換ノード内でパケットを実際に交換する。

【0013】ネットワークを介する経路の決定方法は数多くある。極めて高いスループットを得るために重要なことは、着信パケットをどこに経路指定するかを、交換エレメントが極めて短時間内に判断できなければならぬことである。"High Speed Networking Technology, An Introductory Survey" pp. 88~96, GG24-3816-00 ITSC Raleigh March 1992に述べられているように、最もよく使用される経路指定モードは、ソース経路指定(ソース・ルーティング)およびラベル・スワッピングである。

【0014】・ソース経路指定:

ソース経路指定は、無接続ネットワーク用の分散経路指定の特定の実施態様である。ソース・ノード(またはアクセス・ノード)が、パケットがネットワークを介して取るべき経路を計算する責任を負う。経路指定フィールドが送られたすべてのパケットに付加され、中間ノードがそのフィールドを使ってパケットを宛先に導く。各パケットは、自己完結型ユニットとしてネットワークを介して経路指定され、他のすべてのパケットと無関係である。ソース・ノードに関する重要な点は、各データグラムについて、より短時間で最適の経路を決定することである。

【0015】・ラベル・スワッピング:

ラベル・スワッピングは、接続指向ネットワーク用の分散経路指定の特定の実施態様である。このネットワーク

は、一般的に、何らかの形の「チャネル」を使用するリンク上で、多数の接続(セッション)を多重化する。リンク上を送信される各パケットは、このパケットが属する論理接続を識別する任意の番号を含むヘッダを有する。

【0016】呼出しセットアップおよび資源予約処理は、次のような段階を含む。

- ・接続要求は、ユーザにより起点ネットワーク・アドレスと宛先ネットワーク・アドレスならびにデータ・フロー特性を含むパラメータ・セットを使って指定される。
- ・経路決定は、ソース・ノードによってその局所的経路指定トポロジ・データベースから実現される。
- ・経路予約は、指定されたノードに沿って、特別なメッセージに入れて送出される。このメッセージは、通常は前述のソース経路指定技法に従って経路指定され、トライック・タイプが必要とするサービス・レベルを提供するため、その接続テーブルをセットアップし、その資源を予約する(たとえば、経路の各リンク上の帯域幅予約)よう、ノードに指示する。
- 20 20. テーブル更新は、ネットワーク資源の使用可能性の変化を反映する。その情報は、ネットワーク内のすべてのノードに送られる。

【0017】ラベル・スワッピング技法では、接続テーブルを動的にセットアップし維持する必要がある。これは、新しい接続が確立されまたは古い接続が終了されるとき、テーブルが更新されることを意味する(もちろん、ネットワーク・トポロジのデータベースは完全に独立して維持される)。これは、接続セットアップが比較的たまにしか行われず、あまり時間が重要ではない場合に可能になる。この制限のため、データグラム移送はまったく非効率的である。しかしながら、実時間音声接続のような非常に短いパケットしか必要としない接続は、パケットのオーバヘッドが低いので、この技法により有效地に支援される。接続がいったん確立された後は、パケットが送られるたびに宛先アドレスをヘッダ内に置く必要はない。このパケットのためにどの接続を使用するかを指定するフィールドだけが必要とされる。

【0018】(c) 重要な要件

接続指向経路指定モードと無接続経路指定モードのどちらもサポートする大規模高速パケット交換ネットワークにおける分散経路指定機構では、性能および資源消費量に関して、次のように要約されるいくつかの要件が暗黙に課される。

【0019】・ソース・ノード(またはソース・ノードのために経路計算を行うノード)は、着信パケットをどこに経路指定するかを非常に短期間に決定できなければならない(各接続要求のための最適経路を計算するための計算が十分に早くなければならない)。

・中間ノードにおける交換時間が最短でなければならない(最短処理時間)。

・選択された経路に沿ったネットワーク資源を、最小ホップ・カウントの基準に従って最適化しなければならない。

・ネットワークが過負荷にならないように、制御メッセージをできるだけ制限しなければならない。

【0020】経路選択基準：

ベルマン・フォード・アルゴリズム

パケット・ネットワークにおけるノード間の最適経路を計算するためには、いくつかの適応的経路指定アルゴリズムがある。D. ベルトセカス(Bertsekas)及びR. ガラガー(Gallager)著Data Networks、Prentice-Hall(1987)、米国ニュージャージ州Englewood Cliffs、に記載されたベルマン・フォード・アルゴリズムは、高速パケット交換ネットワークにおいて最も広く使われているものの1つである。このアルゴリズムの目的は、2つのノードをつなぐ最短の長さの経路を見つけることであり、その長さはリンクの混雑を反映するように定義されている。このアルゴリズムは、ソース・ノードから始めて、まず、経路が多くても1つのリンクを含むという制約条件に従って最短経路長を見つける。次に、経路が多くても2つのリンクを含むという制約条件で最短経路長を見つけ、以下同様に続ける。

【0021】・ノードの数をNとする(1、2、…、Nで表す)

・単一方向リンクの数をMとする。あるノードから別のノードに向かう単一方向リンクが多くても1つであり、したがってリンクがその端点で識別できると仮定する。

・ホップの数をhとする。

・ソース・ノード(1)からノード(i)までの最短($\leq h$)経路長さをD(i, h)とする。規約により、すべてのhについてD(1, h) = 0である。

・ノード(i)からノード(j)まで接続するリンクの長さをd_{ij}とする。規約により、ノード(i)がノード(j)に接続されない場合はd_{ij} = ∞ である。

【0022】最初に、i ≠ 1であるすべてのiについて、D(i, 0) = ∞ である。0 ≤ hである連続したhについて、i ≠ 1であるすべてのiについて、D(i, h+1) = m_i n_j [D(j, h) + d_{ji}]、j = 1, …, N

【0023】1つの経路は最高でN-1個のリンクを含む。最悪の場合、アルゴリズムをN-1回反復し、N-1個のノードについて各反復を行わなければならず、また各ノードについて、わずかN-1個の代替案について最小化を行わなければならない。これにより、計算の合計は最悪の場合にはN3回になる。

【0024】ノード(1)からノード(i)までの最短経路長をD(i)とすると、D(1) = 0であり、方程式はh = N-1で収束し、次のように表すことができる。i ≠ 1であるすべてのiについて、

$$D(i) = m_i n_j [D(j) + d_{ji}]$$

$$D(1) = 0$$

【0025】これはベルマン方程式と呼ばれ、ノード(i)からノード(i)までの最短経路長が、ノード(i)の手前にあるノードまでの(最短経路上での)経路長に、経路の最終リンクの距離を加えた合計であることを表す。

【0026】修正ベルマン・フォード方程式

ベルマン・フォード・アルゴリズムで述べたように、パケット・ネットワークにおけるノード間の経路を選択するための主な基準の1つは、最短経路長である。経路長は、顧客の選択基準に依存する。たいていの場合、その長さは、2つの終端ノード間の経路によって課される全伝送遅延の関数である。ほとんどの高速ネットワークでは、そのようなネットワーク内の遅延は最悪の場合でもほとんど常に受け入れられるので、遅延(経路長)は重要な考慮事項ではない。しかし、ホップ・カウントは、所与の経路を実現するために必要な資源の量の直接の尺度であり、したがって経路を選択する際にはかなり重要である。ホップ・カウントは、2ノード間の経路を構成するために使用されるリンク数として定義される。ネットワーク・リンクが混雑すると、ホップ・カウントの多い経路を選択せざるを得なくなることがあるので、選択された経路は必ずしも最低ホップ・カウント経路である必要はない。しかし、法外な量のネットワーク資源が1つの経路にコミットされ、その結果他の経路がさらに混雑し、さらに別の接続用にさらに多いホップ・カウントの経路を選択することを余儀なくされることがあるので、そのようなより長い代替経路を制限なしに成長させることは許されない。それにより、長期間のネットワーク・スループットが悪影響を受けることがある。そこで、問題は、最低のホップ・カウントおよび最低の経路長を有し、法外な量のネットワーク資源を利用しない、起点ノードと宛先ノードの間の経路を選択することである。

【0027】修正ベルマン・フォード・アルゴリズムの目的は、リンクの最低数を定めて所与のソース・ノードと宛先ノードの対の間でソース・ノードと宛先ノードを接続する最低経路長を有する経路を見つけることである。アルゴリズムはソース・ノードから始まり、実現可能な経路が見つかるまでホップ・カウント(リンク数)を増すことによって進行する。このアルゴリズムは、各ホップ・カウントhで、ソース・ノードからちょうどh個のリンクを有する各中間ノードまでの最も短い経路を見つける。

【0028】ソース・ノードからちょうどh個のホップを有するノード(i)までの最短経路をD(i, h)とすると、i ≠ 1であるすべてのiについて、D(i, h) = m_i n_j [D(j, h-1) + d_{ji}]、j = 1, …, N規約により、リンク(ji)が存在しない場合(i = jの場合も含む)は、d_{ji} = ∞ である。

【0029】ソース・ノードをs、宛先ノードをdと

し、アルゴリズムがホップ・カウント h_x で止まるとすると、 $D(d, h_x)$ は、最低ホップ数のすべての実現可能経路のうちで最短の経路長である。

【0030】ベルマン・フォード・アルゴリズムと修正ベルマン・フォード・アルゴリズムとの違いは、ベルマン・フォード・アルゴリズムは最短経路長の経路が見つかるまで続くという点である（各中間ノードへの最短経路長がそれ以上減少できないとき、またはすべてのホップ・カウントを試みたときに止まる）。

【0031】欧州特許出願第93480030.1号明細書（特願平5-90561号）"Methods and Apparatus for Optimum Path Selection in Packet Transmission Networks" と題する欧州特許出願第93480030.1号明細書は、修正ベルマン・フォード・アルゴリズムの改善による起点ノードと宛先ノードとの間の最適経路を選択する方法を開示している。その方法は、所与の起点と宛先の対の間の「主経路」を定義する。主経路は、実現可能な最低ホップ・カウントの経路として定義され、主リンクは、主経路におけるリンクとして定義される。他のすべてのリンクは二次リンクとして定義される。二次経路は、少なくとも1つの二次リンクを含み、最低ホップ・カウントよりも多いホップ・カウントを有する経路である。

【0032】・主経路は、その主リンクがどれも飽和しない場合に、すなわちその事前に割り当てられたトラフィック負荷をどれも越えない場合に、経路として受け入れられる。

・一方、二次経路は、必ずしも全ての主リンクが飽和しない場合で、その二次リンク上の負荷レベルが事前選択されたしきい値（一般に、主リンクとして指定されたリンクのしきい値よりも低い）を下回る場合にだけ経路として受け入れられる。いずれかの二次リンク上でこの負荷しきい値を越えた場合、その二次経路は経路として拒否される。

【0033】この経路選択技法の1つの利点は、経路選択プロセスに最大経路長という制約条件を課すことができることである。すなわち、実現可能な経路を試験して、経路長制約条件を越えたかどうかを判定し、制約条件を越えた場合、経路長制約条件を拒否することができる。そのような制約条件は、経路を実現する際に法外な資源の消費を禁止するために使用することができ、また、低速リンクの回避など特定の等級のサービス要件を課すために使用することもできる。そのため、一般に各接続要求ごとに経路長制約条件を指定しなければならず、また各接続要求ごとに主リンクを個別に決定しなければならない。

【0034】このアルゴリズムは、次のようないくつかのパラメータによって呼び出される。

- ・ソース・ノード・アドレス
- ・宛先ノード・アドレス

・接続のために必要とされる、たとえば次のようなサービス品質。

・最大呼設定延（実時間で処理される接続にとって非常に重要なパラメータ）。

・最大終端間通過延

・最大情報損失、エラー確率等

【0035】このアルゴリズムは、ソース・ノード（またはアクセス・ノード）において、宛先ノードまでの最低ホップ数と最低経路長を決定するために、ネットワーク・トポロジのローカル・コピーを使用する。

【0036】

【発明が解決しようとする課題】特願平5-90561号明細書の経路選択技法は、2つの段階を含む。第1段階では、要求された接続用の主リンクを識別する。最大長の制約条件が課されない場合は、アルゴリズムの第2段階で使うために、ネットワークの任意の2つのノード間の主リンクをあらかじめ計算し記憶しておくことができる。最大長の制約条件が課される場合は、新しい接続要求ごとに主リンクを計算しなければならず、各リンク長を制約条件または以前に受け入れられた主リンク長で軽減された制約条件と比較しなければならない。

【0037】この経路決定技法は、ホップ・カウントと経路長を考慮するだけでなく、経路長に上限値を課すことにより、パケット交換システムの任意のノード間の最適経路を生成するという利点を有する。

【0038】しかし問題は、接続要求ごとに最適経路が選択できるように、経路を十分迅速に計算しなければならない点にある。

【0039】

30 【課題を解決するための手段】本発明の実施例によれば、特願平5-90561号明細書に記載された修正ベルマン・フォード・アルゴリズムによって調査される可能性のあるリンクの数を減らすことにより、パケット・ネットワークにおける起点ノードと宛先ノードの間の最適経路がより効率的に選択される。本発明の経路選択技法は、主リンク識別のための第1段階と、ネットワークにおいて最適の2地点間経路を選択するための第2段階の、2つの段階を含む。

【0040】より詳細には、本発明のアルゴリズムは、40 第2段階におけるネットワーク・リンクのスクリーニング（screening：リンクを調整し選択する処理）を主リンクだけに限定し、二次リンクを除外する。調査するリンクの数をより少なくすることは、経路選択に要する計算時間をかなり減少させるという利点がある。一般に主リンクの数は、調査される可能性のあるリンクの総数に比べて少ないので、試みが成功しない場合に必要とされるさらなる処理時間を制限することができる。

【0041】したがって、本発明は、ネットワークにおける起点ノードと宛先ノードの間の最適経路指定経路を50 決定するためのシステムおよび方法を開示する。前記方

法は、ネットワーク構成を記憶し更新する段階と、第1段階において、前記ネットワークにおける前記起点ノードと前記宛先ノードの間の、最低ホップ・カウントと最低伝送遅延時間を有する主経路を識別し、主リンクと二次リンクを識別する段階と、第2段階において、主経路と主リンクを識別する前記段階に応じて、前記起点ノードと宛先ノードの間の最適経路を決定する段階と、を含み、さらに、前記第2段階において、主リンクだけを使って最適経路を決定する段階と、前記主リンクを使って宛先ノードに到達しないときに、主リンクと二次リンクを使って最適経路を決定する段階とを含むことを特徴とする。

【0042】主リンクおよび二次リンク上で最適経路を決定する前記段階は、前記主リンク上で最適経路を決定する前記段階によって、主リンク上で以前に計算されたリンク利用状況データを使用する。

【0043】

【実施例】図2に示すように、通信システムの典型的なモデルは、専用回線、キャリア提供サービスまたは公衆データ・ネットワークを使用して、高性能ネットワーク(200)を介して通信する、いくつかのユーザ・ネットワーク(212)からなる。各ユーザ・ネットワークは、企業サーバ(213)として使用される大型コンピュータ、LAN(ローカル・エリア・ネットワーク214)に接続されたワークステーションまたはパーソナル・コンピュータを使用するユーザ・グループ、アプリケーション・サーバ(215)、PBX(構内交換機216)またはビデオ・サーバ(217)を相互接続するリンク(211)ならびに通信プロセッサの組として記述される。これらのユーザ・ネットワークは、異なる施設に分散され、広域転送設備を介して相互接続する必要があり、データ転送を編成するために異なる方法が使用できる。あるアーキテクチャは、各ネットワーク・ノードでデータ保全性に関する検査を必要とし、そのため伝送が遅くなる。他のアーキテクチャは、基本的に高速データ転送を追求するもので、最終宛先に向って流れるパケットができる限り最高速度で処理されるように、ノード内の経路指定技法と交換技法が最適化される。本発明は本質的に後者のカテゴリに属しており、より詳細には以下の節で詳しく述べる高速パケット交換ネットワーク・アーキテクチャに属する。

【0044】高速パケット交換ネットワーク：

図2における全体図は、8個のノード(201-208)を備える高速パケット交換伝送システムを示し、各ノードはトランク(209)と呼ばれる高速通信回線によって相互接続されている。ユーザによる高速ネットワークへのアクセス(210)は、周辺に配置されたアクセス・ノード(202-205)を介して実現される。これらのアクセス・ノードは1つまたは複数のポートを含み、各ポートはそれぞれ、標準インターフェースを支

援する外部装置をネットワークに接続し、ネットワークを横切ってユーザのデータ・フローを他の外部装置からまたは他の外部装置へ移送するために必要な変換を実行するためのアクセス点を提供する。たとえば、アクセス・ノード202は、構内交換機(PBX)、アプリケーション・サーバ、および3つのポートを通るハブにそれぞれインターフェースし、隣接する通過ノード201、208および205によってネットワークを介して通信する。

【0045】交換ノード：

各ネットワーク・ノード(201-208)は、着信データ・パケットが発信トランク上で隣接通過ノードに向って選択的に経路指定される、経路指定点を含む。そのような経路指定の決定は、データ・パケットのヘッダ内に含まれる情報に従って行われる。基本的なパケット経路指定機能に加え、ネットワーク・ノードは次のような補助的サービスをも提供する。

- ・ノード内から発するパケット用の指定経路の決定。
- ・ネットワーク・ユーザおよび資源に関する情報の検索や更新などのディレクトリ・サービス。
- ・リンク使用状況情報を含む物理的ネットワーク・トポロジの一貫したビューの維持。
- ・ネットワークのアクセス点での資源の予約。

【0046】各ポートは複数のユーザ処理装置に接続され、各ユーザ装置は、別のユーザ・システムに伝送されるデジタル・データのソース、または別のユーザ・システムから受け取ったデジタル・データを消費するデータ受信側、あるいは通常はその両方を含む。ユーザ・プロトコルの解釈、パケット・ネットワーク(200)上で伝送するために適切にフォーマットされたパケットへのユーザ・データの変換、およびこれらのパケットを経路選択するためのヘッダの生成が、ポートで走行するアクセス・エージェントによって実行される。このヘッダは、制御フィールドと経路指定フィールドからなる。

【0047】・経路指定フィールドは、アドレス指定された宛先ノードにネットワーク(200)を介してパケットを経路指定するために必要な情報をすべて含む。
・制御フィールドは、とりわけ、経路指定フィールドを解釈する際に使われるプロトコルのコード化された識別を含む(ソース経路指定、ラベル・スワッピングなど)。

【0048】経路指定点：

図3は、図2に示したネットワーク・ノード(201-208)中に見られるような、典型的な経路指定点(300)の全体的ブロック図を示す。経路指定点は、高速パケット・スイッチ(302)を含み、経路指定点に到達したパケットがそこに入る。そのようなパケットは以下のようにして受信される。

- ・高速伝送リンク(303)を経てトランク・アダプタ(304)を介して、他のノードから受信する。

・ポート（301）と呼ばれるアプリケーション・アダプタを介して、ユーザから受信する。

【0049】アダプタ（304、301）は、パケット・ヘッダ内の情報を使用して、どのパケットをスイッチ（交換網：302）によってローカル・ユーザ・ネットワーク（307）に向けてまたはノードから出る伝送リンク（303）に向けて経路指定するかを決定する。アダプタ（301と304）は、スイッチ（302）に入る前または後にパケットを待ち行列に入れるための待ち行列回路を含む。

【0050】経路推定制御装置（305）は、ユーザによって指定された所与の1組のサービス品質を満たし、通信経路を完成させるために用いられるネットワーク資源の量を最小にするために、ネットワーク（200）を介する最適経路を計算する。次に、経路指定点で生成されたパケットのヘッダを構築する。最適化基準には、中間ノードの数、接続要求の特徴、経路内のトランクの容量と使用度が含まれる。経路指定のために必要な、ノードおよびノードに接続された伝送リンクに関する情報はすべて、ネットワーク・トポロジ・データベース（306）内に含まれる。定常状態のもとでは、すべての経路指定点は同じネットワークのビューを有する。ネットワーク・トポロジ情報は、ネットワークに新しいノードが追加されて新しいリンクが活動化されたとき、リンクまたはノードがドロップされたとき、あるいはリンクの負荷が著しく変化したときに更新される。そのような情報は、その資源が接続されたネットワーク・ノードから発し、制御メッセージによって他のすべての経路制御装置との間で交換され、経路計算に必要な最新のトポロジ情報を提供する（そのようなデータベースの更新は、ネットワークのエンド・ユーザ間で交換されるデータ・パケットと非常に類似したパケット上で搬送される）。継続的な更新を通じてネットワーク・トポロジがあらゆるノードで最新の状態に維持されるので、エンド・ユーザの論理接続（セッション）を乱さずに、動的なネットワークの再構成が可能になる。

【0051】パケット経路指定点への着信伝送リンクは、ローカル・ユーザ・ネットワークの外部装置からのリンク（210）、または隣接するネットワーク・ノードからのリンク（トランク：209）を含む。どちらの場合でも、経路指定点は同じように動作して、パケット・ヘッダ内の情報によって指示されるように、各データ・パケットを受け取り、別の経路指定点に転送する。高速パケット交換ネットワークは、単一パケットの伝送継続中以外はどの伝送機構やノード機構をも通信経路専用にせずに、任意の2つのエンド・ユーザ・アプリケーション間の通信を可能にするように働く。このようにして、各通信経路ごとに伝送リンクを専用にする場合よりも、ずっと多くのトラフィックが搬送できるように、パケット・ネットワークの通信機構の利用度が最適化され

る。

【0052】リンク特性：
トポロジ・データベース

図4は、トポロジ・データベース306に記憶された情報の一部を表形式で示したものである。ネットワーク内の各リンクの種々な特性が、データベースにリストされている。本発明に関して、これらの特性のうちのいくつかだけを検討する。

【0053】-負荷しきい値

10 予想されるとおり、伝送リンクの重要な特性の1つとして負荷しきい値が使用可能である。さらに、そのような伝送装置は、妥当な伝送特性を維持しなければならない場合、その理論上の最大負荷（帯域幅）の一部分までしか負荷をかけることができないことはよく知られている。そのような伝送装置の負荷しきい値は、量 C_{k1} （ノード（k）とノード（1）の間の伝送リンクの有効負荷能力）で表すことができる。リンクがルート内の主リンクとして選択されるか、それともルート内の二次リンクとして選択されるかに応じて、各伝送リンクごとに2つの異なる負荷しきい値 $L(P)$ および $L(S)$ が定義される。

【0054】-主リンクと二次リンク

主リンクは、起点ノードと宛先ノードの間で実現可能な最低ホップ・カウント経路である主経路の脚部として定義される。ホップ・カウント h は、単にその経路における伝送リンクの数である。他のすべてのリンクは、二次リンクとして定義される。起点ノードと宛先ノードの間のあらゆる非最低ホップ・カウント経路は二次経路と呼ばれ、少なくとも1つの二次リンクを常に含む。

30 30 【0055】ノード間の最適経路を決定する際には、二次経路よりも主経路が好まれる。また、必要な計算時間は、二次経路よりも主経路の方がずっと短い。しかし、主経路がすでに完全に負荷がかかっていて使用できない場合は、二次経路を選ぶことができる。そのような二次経路を識別するために、二次リンクごとに、同じリンクの対応する主負荷しきい値 $C_{k1,P}$ よりも小さい負荷しきい値 $C_{k1,S}$ が定義される（ $C_{k1,P} \geq C_{k1,S}$ ）。このような理由で、図4の表には伝送リンクごとに異なる2つの負荷しきい値を含み、そのリンクが計算される経路において主リンクである場合は一方を使用し、計算される経路において二次リンクである場合は他方を使用する。

40 40 【0056】-合計割振り負荷

表には、リンクごとに、合計割振り負荷 $T(AL)$ が含まれている。本明細書において値

【数1】

\hat{C}

は以降 C_{k1} と記載する。この値は、経路が前に計算されたために伝送リンク $k1$ （ノード k とノード 1 の間の伝送リンク）にすでに割り振られている合計負荷

50 （または帯域幅）を表す。このすでに割り振られた負荷

わる。宛先ノードに到達した場合(602)は、ホップ・カウントと経路長を使って実現可能なリンクのリストを後戻り(バックトラック)して(603)、最短の主経路を見つける。主経路はすべて同じ最低ホップ・カウントを有する。図6のプロセスはステップ604で終わる。

【0068】2. 従来型バージョン

図7は、最適経路を計算する第2段階のより詳細なフローチャートである。第2段階は、本発明によれば加速型バージョン(701, 704)と従来型バージョン(703, 704)を含む。従来型バージョンは、加速型のオプションとしてルーチンA1で所望経路の宛先ノードに到達することができなかつたときに用いられるプロセスである。第2段階は、ステップ700において加速型バージョンを開始する。実現可能な主経路を見つけるためにすべての主リンクを探索した後、ルーチンA1(701)の終りに宛先ノードに到達しない場合(702)は、ルーチンA2(703)が実行される。ルーチンA2は、特願平5-90561号明細書に記載された従来の方法に従って動作する。主リンクと二次リンクの両方が考慮され、二次リンクは、新しい接続を追加しても二次リンクの負荷しきい値Ck1,Sを越えない場合に実現可能なものとして受け入れられる。しかし、ルーチンA2は、ステップ701でルーチンA1によって主リンクに関して以前に行われたCハットk1(2)に関するすべての検査および計算を利用できる。そのような理由で、2つのルーチンA1(701)とA2(703)を連鎖しても、本発明の第2段階は、特願平5-90561号明細書に記載された第2段階と比べてあまり劣化しない。ステップ704で、実現可能なリンクのリストがホップ・カウントおよび経路長を使ってバックトラックされ、最低ホップ・カウントの最短経路を見つける。第2段階はステップ705で終わる。

【0069】注:

・第1段階のルーチンA、加速型の第2段階のルーチンA1、および従来型の第2段階のルーチンA2で使われる修正ベルマン・フォード・アルゴリズムは、特願平5-90561号明細書の図8に記載されたものと同じである。

・第1段階のルーチンB、加速型および従来型の第2段階のルーチンBで使用されるバックトラック手順は、特願平5-90561号明細書の図9に記載されたものと同じである。

【0070】-探索アルゴリズム

1. 加速型バージョン(ルーチンA1)

図8は、最適経路の加速型探索を実行するために使われる修正ベルマン・フォード・アルゴリズムの詳細なフローチャートを示す。ステップ800で始まり、まずステップ801で、次のノードを起点ノード(i)にセットする。ステップ802で、次のノードに関するデータを

トポロジ・データベース(306)から検索する。ステップ803で、そのノードから出る次のリンクに関するデータを、同じトポロジ・データベース(306)から得る。判断点804で、そのリンクが主リンクかどうかを判定する。

【0071】・リンクが主リンクの場合は、判断点805に移り、

・ステップ812で、累積負荷Cハットk1(2)を計算する。次にCハットk1(2)をリンクの主しきい値Ck1,Pと比較する。

・累積負荷が主しきい値と等しいかまたはそれよりも小さい場合(Cハットk1(2) ≤ Ck1,P)は、判断点806で累積経路長または累積経路遅延を試験する。

・判断点805で累積負荷が主しきい値よりも大きいと判定された場合(Cハットk1(2) ≥ Ck1,P)は、ステップ808で次のリンクを得る。

・判断点804でそのリンクが主リンクではないと判定された場合は、そのリンクは考慮されず、ステップ808で次のリンクを得る。

20 【0072】判断点806で、この点までの累積経路長Σdk1を、最大経路長PTと比較する。

・累積経路長Σdk1がPTよりも小さい場合は、ホップ・カウントhkおよび累積経路長dk1をリストに保管し(807)、必要ならば次のホップで反復を続けるために到達ノードも保管する。

・累積経路長Σdk1がPTと等しいかまたはそれよりも大きい場合は、ステップ807を迂回し、このリンクのデータはリストに加えられない。

30 【0073】いずれの場合も、次に判断点808に入つて、このノードから出るリンクが他にもあるかどうかを判定する。他にもリンクがある場合には、ステップ803で、次のリンクを得てプロセスを続ける。このノードから出るリンクがもうない場合、判断点809でそのノードが宛先ノード(j)であると判定される。そのような場合、プロセスは完了し、ステップ811で終了する。このノードが宛先ノード(j)ではない場合は、判断点810で、スクリーニングすべきノードが他にもあるかどうかを判定する。スクリーニングすべきノードがまだある場合は、ステップ802で次のノードを得てプロセスを続ける。スクリーニングすべきノードがもうない場合は、加速型探索は失敗し、ステップ900で図9に記載された従来型探索を開始する。

40 【0074】図8の手順が完了し宛先ノード(j)に到達すると、最低ホップ・カウントhkおよび経路長dk1のリストが使用可能になる。先に考察したように、次にルーチンBはこのリストをバックトラックして、最小経路遅延ならびに最低ホップ・カウントを有する経路を識別する。これが、図2のシステムによってこの接続に関わるパケットを送る際に使用される最適経路である。

50 【0075】注: 主経路を識別するために修正ベルマン

・フォード・アルゴリズムを使用するときは、特願平5-90561号明細書の図10に示されたものと同じフローチャートを使用する。

【0076】2. 従来型バージョン（ルーチンA2）

図9は、最適経路に関する従来型探索を実行するのに使用される修正ベルマン・フォード・アルゴリズムの詳細なフローチャートを示す。ステップ900で始まり、まずステップ901で、次のノードを起点ノード（i）にセットする。ステップ902で、次のノードに関するデータをトポロジ・データベース（306）から検索する。ステップ903で、そのノードから出る次のリンクに関するデータを同じトポロジ・データベースから得る。判断点904で、そのリンクが主リンクかどうかを判定する。

【0077】・そのリンクが主リンクである場合は、判断点905で、累積負荷Cハットk1(2)をリンクの主しきい値Ck1,Pと比較する。この累積負荷は加速型探索アルゴリズムで以前に計算してあり、そのためここで再使用できる。

・累積負荷が主しきい値と等しいかまたはそれよりも小さい場合は、判断点907で累積経路遅延を試験する。

・判断点905で累積負荷が主しきい値よりも大きいと判定された場合は、ステップ909で次のリンクを得る。

==>主リンク処理用の疑似コード

```
IF (k, l) ∈ PL /* ルーチンA1 から Cハットk1(2) が 使用可能 */
  IF Cハットk1(2) < Ck1,P
    IF Σ dkl < PT
      /*段階A2の実行を続ける*/
```

【0078】・判断点904で、リンクが二次リンクと判定された場合は、

- ・ステップ913で累積負荷Cハットk1(2)を計算する。

- ・次に判断点906で、累積負荷Cハットk1(2)をリンク二次しきい値Ck1,Sと比較する。

- ・累積負荷が二次しきい値と等しいかまたはそれよりも小さい場合は、判断点907で累積経路遅延を試験する。

- ・判断点906で累積負荷が二次しきい値よりも大きいと判定された場合は、ステップ909で次のリンクを得る。

==>二次リンク処理用の疑似コード

【数2】

```
IF (k, l) ∈ PL
  /* Cハットk1(2)を計算しなければならない*/
  Cハットk1(2)を計算。 /*Cハットk1(l)とcを含む複雑な計算*/
  IF Cハットk1(2) < Ck1,S
    IF Σ dkl < PT
```

…/*段階A2の実行を続ける*/

判断点907で、この点までの累積経路長を最大経路長PTと比較する。

【0079】・累積経路長がPTよりも小さい場合は、ステップ908で、ホップ・カウントhkと累積経路長dklをリストに保管する。

・累積経路長がPTと等しいかまたはそれよりも大きい場合は、ステップ908を迂回し、このリンクに関するデータはリストに加えられない。

10 【0080】いずれの場合も、次に判断点909に移って、このノードから出るリンクが他にもあるかどうかを判定する。リンクが他にもある場合は、ステップ903で次のリンクを得てプロセスを続ける。このノードから出るリンクがもうない場合は、判断点910に移って、そのノードが宛先ノード（j）かどうかを判定する。そのノードが宛先ノード（j）である場合は、プロセスは完了しステップ912で終了する。このノードが宛先ノード（j）ではない場合は、ステップ911で、スクリーニングすべきノードが他にもあるかどうかを判定する。

20 【0081】図9の手順が完了すると、最大ホップ・カウントhkおよび経路長dklのリストが使用可能になる。先に考察したように、次にルーチンBはこのリストをバックトラックして、最低経路遅延ならびに最低ホップ・カウントを有する経路を識別する。これが、図2のシステムによって接続に関わるパケットを送る際に使用される最適経路である。

【0082】実施態様の例：

－代表的ネットワークにおける経路探索

図10は、本発明によって実現される改善点を例示するために用いられる代表的ネットワークを示す。これは、33個のノードと、たとえばT1（1.5Mbps）、T3（4.5Mbps）、E1（2Mbps）、E3（3.4Mbps）、OC3、STM1（155Mbps）など様々な処理能力の126個の單一方向高速伝送リンク（63トランク）とを含むパケット交換ネットワークである。平均接続性は1ノードあたり2トランクであり、最低1トランク（オスロ、エジンバラ、ダブリン、サンノゼ、ボストン、ワルシャワ）、最高11トランク（パリ）である。本発明の目的は、ソース・ノードと宛先ノードの間の最適経路を探索する際に、調査される可能性のあるリンクの数を減らすことである。この例では、ソースはロサンゼルス（USA）であり、宛先はミラノ（イタリア）である。経路決定手順の第1段階では、3の最低ホップ・カウントを有する4つの主経路を識別する。

40 【数2】

・ロサンゼルス-ロンドン-チューリッヒ-ミラノ間

- ・ロサンゼルス-ロンドン-パリ-ミラノ間
- ・ロサンゼルス-ラ-レイ-ニース-ミラノ間
- ・ロサンゼルス-ニューヨーク-パリ-ミラノ間

【0083】第1段階が首尾よく達成されると、最適経路を決定するために第2段階が実行される。

・加速型バージョン

図11は、本発明のルーチンA1の対象として修正ベルマン・フォード・アルゴリズムによって調べられるリンク数を示す。経路長および実現可能性を決定するために、主リンクだけを考慮する。この例では、ネットワークは、ロサンゼルス-ロンドン間、ロサンゼルス-ラ-レイ間、ロサンゼルス-ニューヨーク間、ラ-レイ-ニース間、ニース-ミラノ間、ロンドン-チューリッヒ間、ロンドン-パリ間、パリ-ミラノ間、チューリッヒ-ミラノ間およびニューヨーク-パリ間の10個のリンクに減少する。

【0084】・従来型バージョン

図12は、特願平5-90561号明細書に記載された従来型の第2段階のルーチンAで使用される修正ベルマン・フォード・アルゴリズムによって調査されるリンクの数を示す。探索は主リンクだけに限定されず二次リンクも含む。たとえば、モスクワには3ホップで到達でき

る。最適経路の探索では、このアルゴリズムによって、ニューヨーク-ベルリン間およびベルリン-モスクワ間の二次リンクが調査される。この事例では、ネットワークの80%に相当する54個のリンクが調査された。

【0085】-性能

新しい経路選択手順によってもたらされる改善点を元のバージョンと比較して定量化するために、シュミレーションを行って、AIX Riscシステム/6000ワークステーションで実行される(XLCでコンパイルされた)経路選択コードの性能を評価した。プログラム中で実行されたすべての命令およびその使用頻度のリストを出力で提供するAIXツールを使用した。

【0086】シュミレートしたネットワークは、図10に記載したネットワークであり、33個のノードと126個の単一方向リンク(63トランク)を含む。3つの試験が実行され、それぞれ最初は特願平5-90561号明細書に詳述された仕様(従来型)で実行し、次に本発明の経路決定手順(加速型)を実行した。計算の結果を下記の表で比較する。実行された命令の数で見た性能

20 利得は約15%である。

【0087】

【表1】

試験	ホップ数	第1段階の調査リンク数	実行命令		利得
			従来型	加速型	
1	2	22	84. 005	73. 604	13%
2	3	54	189. 331	156. 955	17%
3	5	63	241. 341	199. 723	17%

【0088】性能利得は、特に、より長い経路の場合、またたとえば浮動小数点処理能力のないプロセッサなどかなり劣る計算処理能力のプロセッサを使用するときは、50%以上にも達する。

【0089】注:

・これらの結果は経路選択手順の2つの段階を考慮しており、第1段階は変更されないままであり、第2段階は本発明に従って新しいバージョンに更新されている。

・第2経路は第1経路と比べてより多くの浮動小数点計算を有するので、浮動小数点装置のないハードウェア・プラットフォームでは利得が大きくなる。たとえば、インテルのプロセッサ960CF(浮動小数点装置なし)では、平均4ホップの経路について利得は50%と評価される。

【0090】付録:

-定義

- ・Nは、ネットワーク内のノードの合計数である。
- ・hは、アルゴリズムの反復指標であり、ホップ・カウントと等しい。

・hxは、経路長制約条件を満たす、ソース・ノードから宛先ノードまでの最低ホップ経路のホップ・カウントである(アルゴリズムで実現可能経路が見つからない場合は、hx=0)。

・hkは、ソース・ノードからノード(k)までのホップ数である。

・PTは、起点(i)ノードと宛先(j)ノードの間の最大経路長しきい値である(要求接続によって許容される最大遅延)。

・dklは、ノード(k)とノード(l)の間のリンク(もしあれば)の長さ(または遅延)である。ノードkとノードlの間にリンクがない場合、またはk=lの場合、dkl=∞である。

・Cklは、ノード(k)とノード(l)の間のリンクの有効負荷(または帯域幅)能力(負荷しきい値)である。

・Ckl,Pは、経路における主リンクとしてのリンク(kl)の負荷しきい値L(P)である。

50 ・Ckl,Sは、経路における二次リンクとしてのリンク(k

1)の負荷しきい値L (S) である。

- ・Cハットk1は、ノード(k)とノード(1)の間のリンクの合計割振り負荷T (AL) である(経路が以前に計算済みのために、そのリンクにすでに割り振られている負荷の合計)。
- ・リンク(k1)上の現割振り負荷
- ・Cハットk1(1)は、要求接続を考慮に入れない割振り負荷である。
- ・Cハットk1(2)は、要求接続の後の割振り負荷である。
- ・D(1, h)は、ソース・ノードからノード(1)までの、ちょうどhホップの最小経路長(遅延)である(そのような経路が存在しない場合は∞である)。

【0091】まとめとして、本発明の構成に関して以下の事項を開示する。

【0092】(1) 伝送リンク(209)と相互接続された複数のノード(201~208)を含むパケット交換ネットワーク(200)におけるネットワーク・ノード(300)であって、データ・パケットを受信し送信する手段(301、302、304)と、ネットワーク構成を記憶し更新する手段(306)と、前記ネットワークにおける起点ノードと宛先ノードの間において、最低ホップ・カウントと最低伝送遅延時間Σd_{k1}を有する主経路を識別し、主リンクと二次リンクを識別する手段と、主経路と主リンクを識別する前記手段に応答して、前記起点ノードと宛先ノードの間の最適経路を決定する手段とを含み、前記最適経路を決定する前記手段が、主リンクだけを使って最適経路を決定する手段を含むことを特徴とするネットワーク・ノード。

(2) 前記最適経路を決定する前記手段が、前記主リンクだけを使って宛先ノードに到達しないときは、主リンクおよび二次リンクを使って最適経路を決定する手段を含む、上記(1)に記載のネットワーク・ノード。

(3) 主リンクおよび二次リンクを使って最適経路を決定する前記手段が、前記主リンク上で最適経路を決定する前記手段によって主リンク上で以前に計算済みのリンク使用状況データを利用することを特徴とする、上記(2)に記載のネットワーク・ノード。

(4) 前記最適経路における伝送遅延を、事前選択したしきい値PTに制限する手段を含むことを特徴とする、上記(1)ないし(3)のいずれか1に記載のネットワーク・ノード。

(5) 前記起点ノードと宛先ノードの間の最適経路を決定する前記手段が、修正ベルマン・フォード・アルゴリズムを実行する手段を含む、上記(1)ないし(4)のいずれか1に記載のネットワーク・ノード。

(6) 伝送リンク(209)と相互接続された複数のノード(201~208)を含むパケット交換通信ネットワーク(200)におけるアクセス・ノード(300)において、起点ノードと宛先ノードの間の最適経路を決

- 定するための方法であって、ネットワーク構成を記憶し更新するステップと、第1段階(503)で、前記ネットワークにおいて前記起点ノードと前記宛先ノードの間において最少ホップ・カウントと最低伝送遅延時間Σd_{k1}を有する主経路を識別し、主リンクと二次リンクを識別するステップと、第2段階(504)で、主経路と主リンクを識別する前記ステップに応答して、前記起点ノードと宛先ノードの間で最適経路を決定するステップを含み、最適経路を決定する前記第2段階(504)が、主リンクだけを使って最適経路を決定するステップ(701、704)を含むことを特徴とする方法。

(7) 最適経路を決定する前記第2段階(504)が、前記主リンクだけを使って宛先ノードに到達しないとき(702)は、主リンクおよび二次リンクを使って最適経路を決定するステップ(703、704)を含む、上記(6)に記載の方法。

(8) 主リンクおよび二次リンクを使って最適経路を決定する前記ステップが、前記主リンク上で最適経路を決定する前記ステップによって主リンク上で以前に計算済みのリンク使用状況データを利用することを特徴とする、上記(7)に記載の方法。

(9) 前記最適経路における伝送遅延を、事前選択したしきい値PTに制限するステップ(505)を実行するステップを含む、上記(6)ないし(8)のいずれか1に記載の方法。

(10) 前記起点ノードと宛先ノードの間の最適経路を決定する前記ステップが、修正ベルマン・フォード・アルゴリズムを含むことを特徴とする、上記(6)ないし(9)のいずれかに記載の方法。

【0093】

【発明の効果】本発明に寄り、調査の対象となるリンクの数を軽減することができ、所定のノード間での最適経路を指定する際の処理時間を飛躍的に向上させることができる。

【図面の簡単な説明】

【図1】必要な処理時間(1秒あたりの命令数)を、高速ネットワークによって支援される異なるライン・スループットの関数として示す図である。

【図2】本発明による、アクセス・ノードとノードを含む高速パケット交換ネットワークの典型的なモデルを示す図である。

【図3】本発明による高速経路指定点を記述する図である。

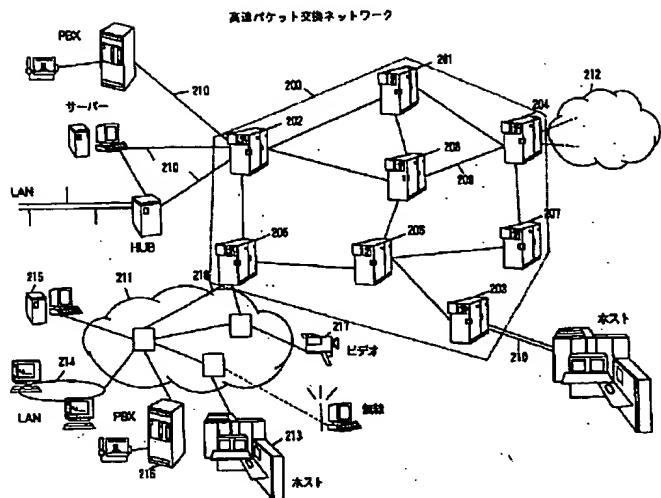
【図4】各経路指定点におけるトポロジ・データベースの一部分を示す図である。

【図5】本発明による経路決定手順の全体的フローチャートである。

【図6】本発明による加速型の第2段階の詳細なフローチャートである。

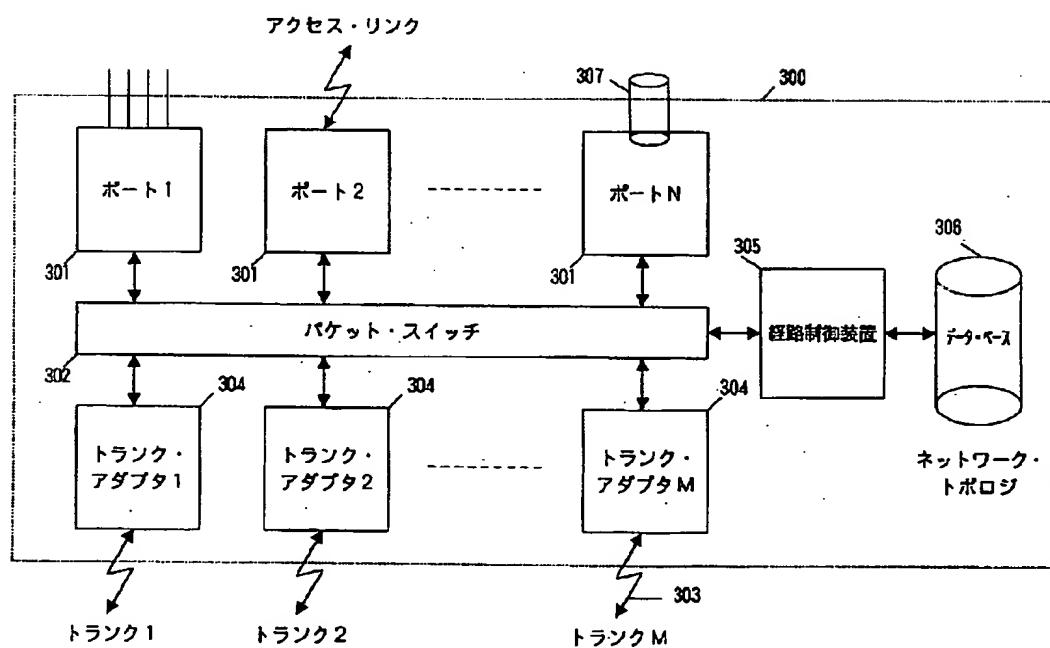
【図7】本発明による従来型の第2段階の詳細なフロー

【図2】



【図3】

高速パケット交換ノード

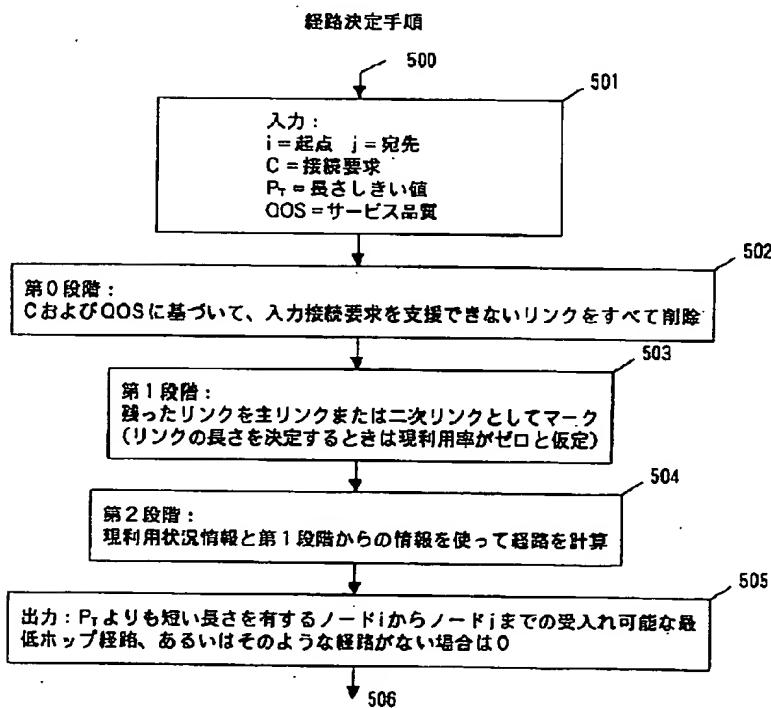


【図4】

トポロジ・データベースリンク特性データ

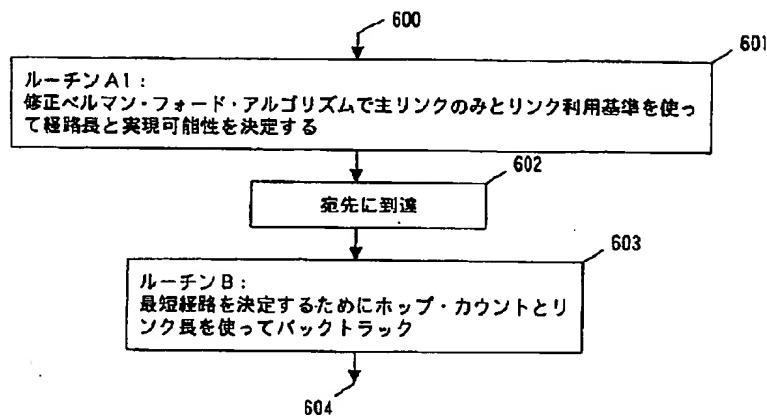
特 性		リンク値			
名 前	説 明	A	B	...	N
L (P)	主リンクの場合の負荷しきい値	$C_{A,P}$	$C_{B,P}$...	$C_{N,P}$
L (S)	二次リンクの場合の負荷しきい値	$C_{A,S}$	$C_{B,S}$...	$C_{N,S}$
T (AL)	合計割振り負荷	\hat{C}_A	\hat{C}_B	...	\hat{C}_N
QOS	サービス品質パラメータ	QOS_A	QOS_B	...	QOS_N

【図5】



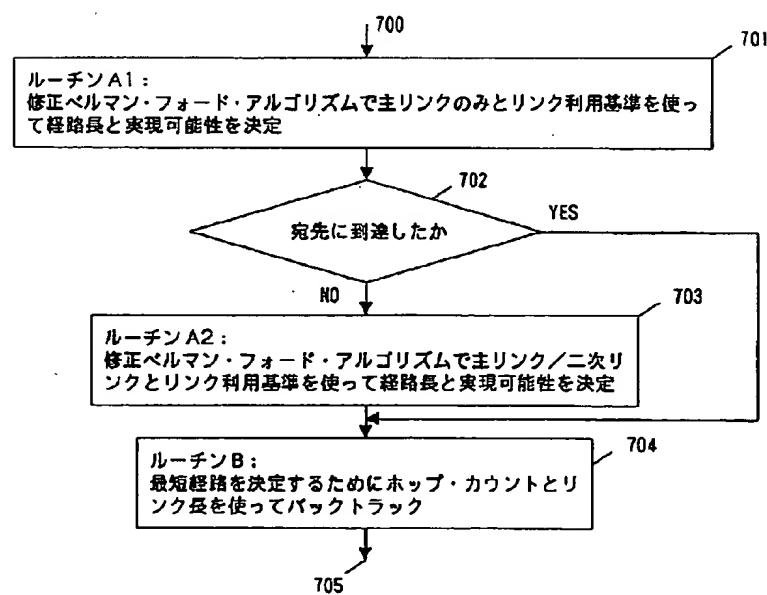
【図6】

第2段階：加速型バージョンで最適経路を決定する



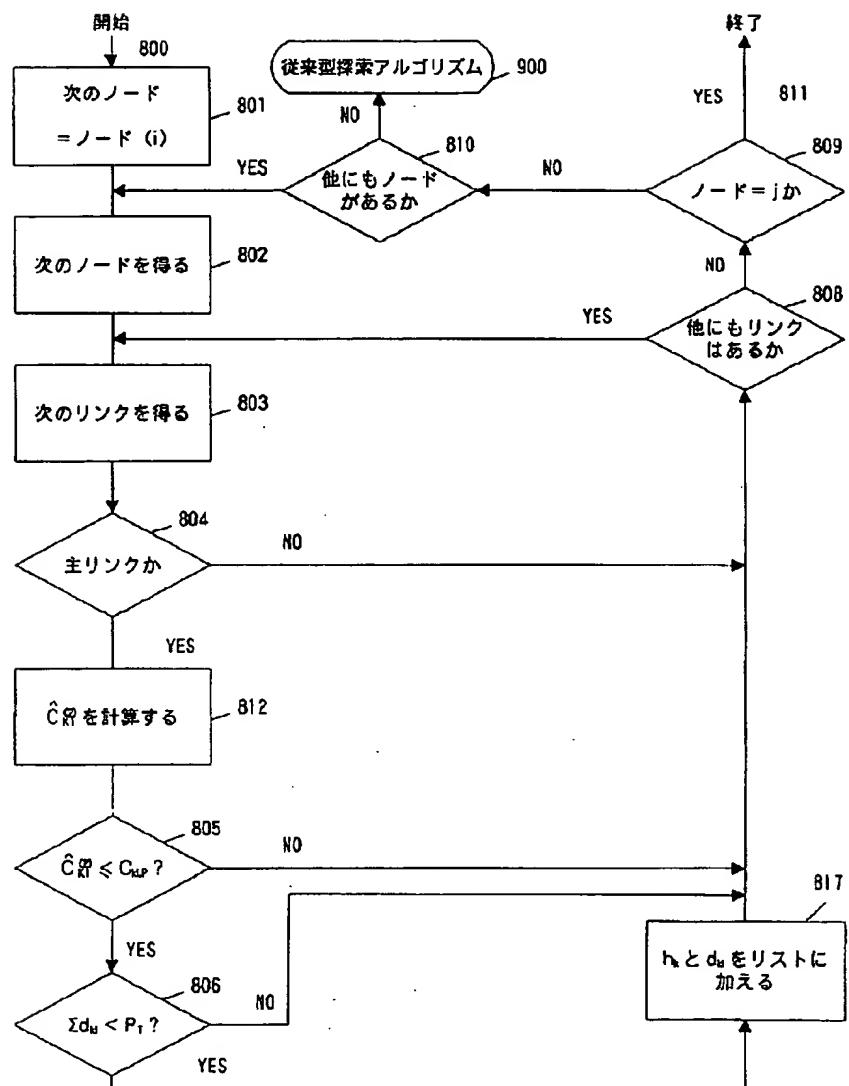
【図7】

第2段階：従来型バージョンおよび加速型バージョンで最適経路を決定する



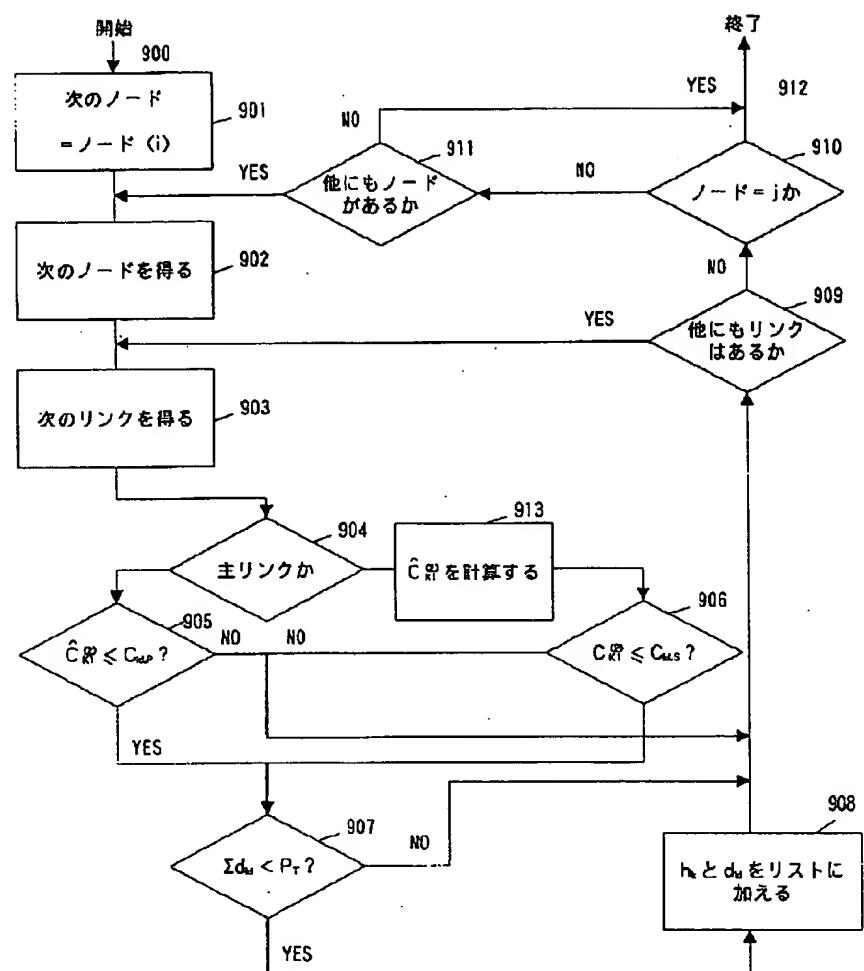
【図8】

加速型探索アルゴリズム（ルーチンA1）

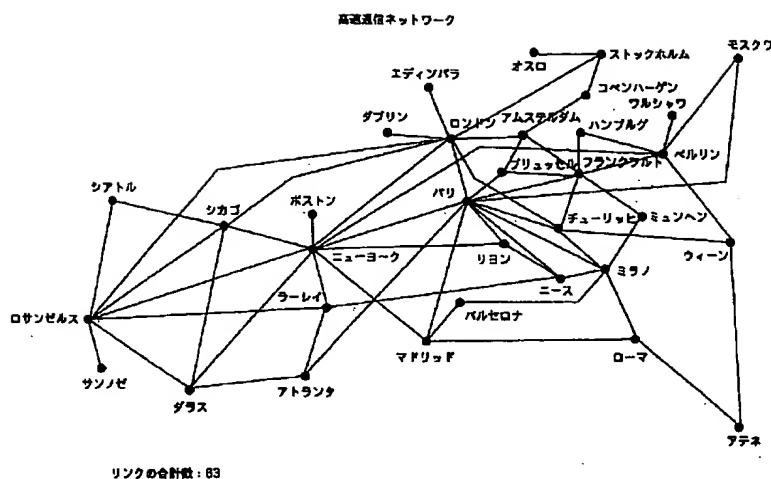


[图 9]

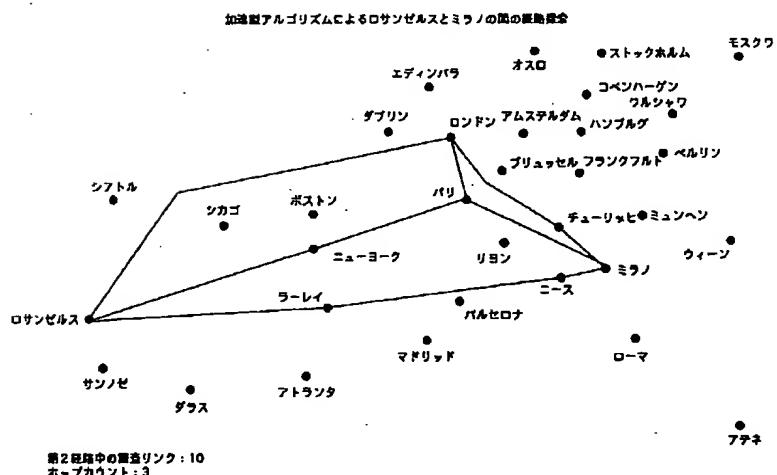
従来型探索アルゴリズム（ルーチンA2）



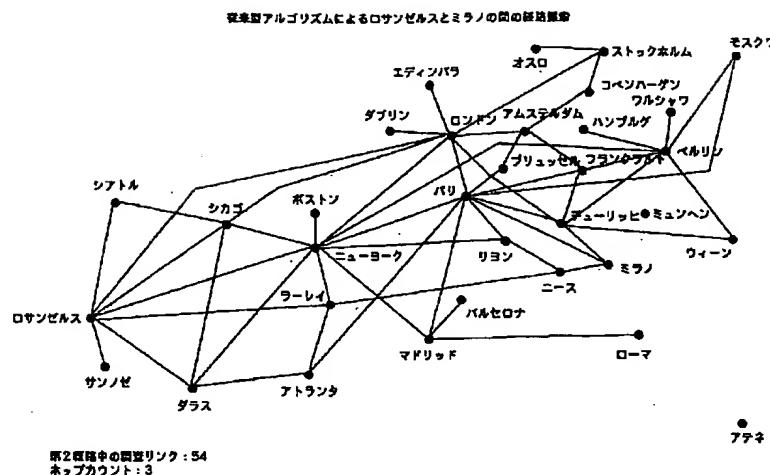
【图 10】



【図11】



【図12】



フロントページの続き

(72)発明者 モーレル・オリヴィエ
 フランス06110 カンヌ リュ・フォルヴ
 イユ 15